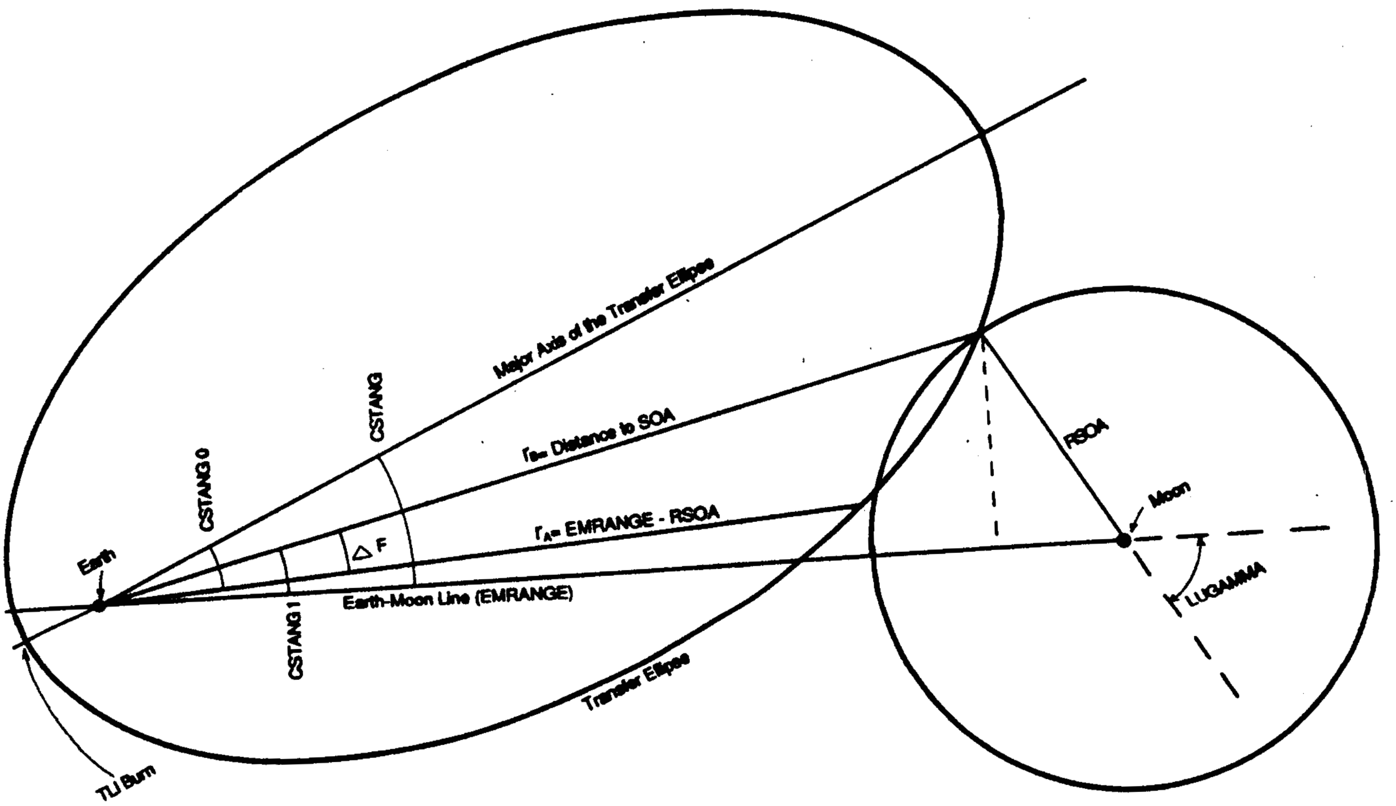
LBSS

LLOFX
Earth Orbit to Lunar Orbit Delta V Estimation Program User and Technical Documentation



(NASA-CR-172091) LLOFX EARTH ORBIT TO LUNAR ORBIT DELTA V ESTIMATION PROGRAM USER AND TECHNICAL DOCUMENTATION (Eagle Engineering) 44 p CSCL 22C

N89-13443

Unclas

G3/13 0179853

NASA Contract Number NAS9-17878
EEI Report Number 88-212
April, 1988

EAGLE

LLOFX

User and Technical Documentation

Eagle Engineering, Inc.

April, 1988

TABLE OF CONTENTS

PAGE

1.0 Introduction

The program LLOFX calculates in-plane trajectories from an Earth-orbiting space station to Lunar orbit in such a way that the journey requires only two delta-v burns (one to leave Earth circular orbit and one to circularize into Lunar orbit). The program requires the user to supply the space station altitude and Lunar orbit altitude (in kilometers above the surface), and the desired time of flight for the transfer (in hours). It then determines and displays the trans-Lunar injection (TLI) delta-v required to achieve the transfer, the Lunar orbit insertion (LOI) delta-v required to circularize the orbit around the Moon, the actual time of flight, and whether the transfer orbit is elliptical or hyperbolic. Return information is also displayed. Finally, a plot of the transfer orbit is displayed.

2.0 Principle Behind the Program

Calculation of the trajectory takes advantage of the fact that the Moon travels at great velocity in orbit about the Earth (1.02 kilometers per second). The vehicle's circular orbit about the Earth is turned into an elliptical transfer orbit that intercepts the Moon's orbit. This transfer orbit is rotated ahead of the Earth-Moon line in such a way that, as the vehicle enters the Moon's sphere of action (SOA) ahead of the Moon, the high velocity of the Moon in the direction of the vehicle causes the vehicle to appear to be headed back toward the Moon (from a Lunar point of view). This program identifies the eccentricity, size, and rotation of the transfer ellipse or hyperbola that causes the velocity vector of the vehicle (in Lunar coordinates) to correspond to an orbit passing in front of the Moon with a perigee at the Lunar orbit altitude supplied by the user.

3.0 Description of the Process

Given the altitude of the space station circular orbit, the program calculates circular velocity. Through a process of iteration, velocity is added to this in small increments (delta-v) so that the orbit becomes elliptical or hyperbolic. Assuming that the burn occurs at the far side of the Earth on the Earth-Moon line, such an ellipse/hyperbola will be symmetrical along the Earth-Moon line, as defined at the time of SOA penetration (see Figure 1). The velocity is increased until this trajectory orbit's apogee is beyond the Moon's SOA. (A hyperbolic orbit meets this condition by definition).

Initially, the program identifies the vehicle's velocity vector (in Lunar coordinates) at a point on the transfer orbit such that the distance from the Earth to that point equals the distance from the Earth to the Moon's SOA, measured along the Earth-Moon line. Note that because of the Moon's motion around the Earth, and because we are using Lunar coordinates, the velocity vector points away from the transfer orbit.

The position on the SOA is determined at which the velocity vector identified above would correspond to an orbit passing in front of the Moon with a perigee equal to the user supplied lunar orbit altitude (see Figure 2). The position is identified by an angle centered at the Moon (LUGAMMA), measured up from the Earth-Moon line. The ellipse/-hyperbola is rotated through an angle called the coast angle (CSTANG) such that it intersects the SOA at this position. (The vehicle physically performs this by causing the TLI burn to occur at an angle past the Earth-Moon line equal to the coast angle. See Appendix D for a description of how the coast angle is calculated.)

This point of intersection of the SOA and the transfer orbit occurs further along the transfer orbit than the point at which the initial velocity vector was identified. The velocity vector at this new point is different from the originally calculated velocity vector (see Figure 3). This new velocity vector (B in the diagram) is not pointing in a direction that will allow interception with Lunar orbit perigee. A new position on the SOA must be determined to allow this condition to be met (B' in the diagram). Figure 4 shows the calculations required to determine this new position (LUGAMMANEW) given the velocity vector's x- and y-components and the flight path angle (ANGMOMGAM). (See Appendix E for a description of how ANGMOMGAM is calculated).

The steps described in the previous paragraph must be iterated until LUGAMMANEW converges (to within 0.02 radians). The resulting coast angle describes the point in Earth orbit at which to perform the TLI burn.

The program then adds the time of flight between SOA and Lunar orbit to the time of flight between TLI and SOA to get total time of flight. If the total flight time does not fall within one hour of the desired time of flight, TLI delta-v is adjusted by 1/10th the percentage of error between actual and desired time, and the entire process is begun anew. Otherwise, the program displays the transfer orbit properties.

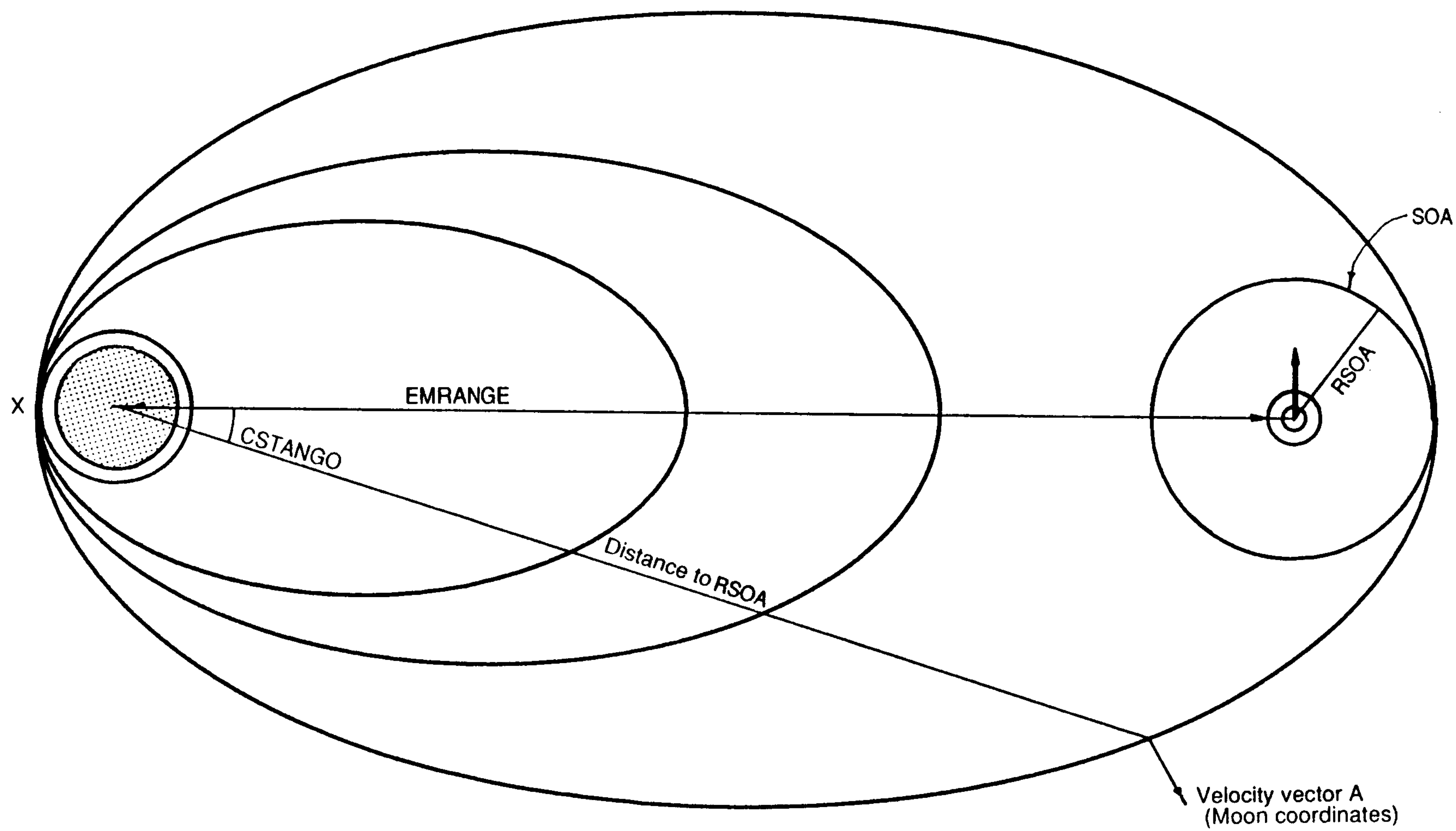

SOA
RSOA
EMRANGE
CSTANGO
Distance to RSOA
X
Velocity vector A
(Moon coordinates)

Figure 1

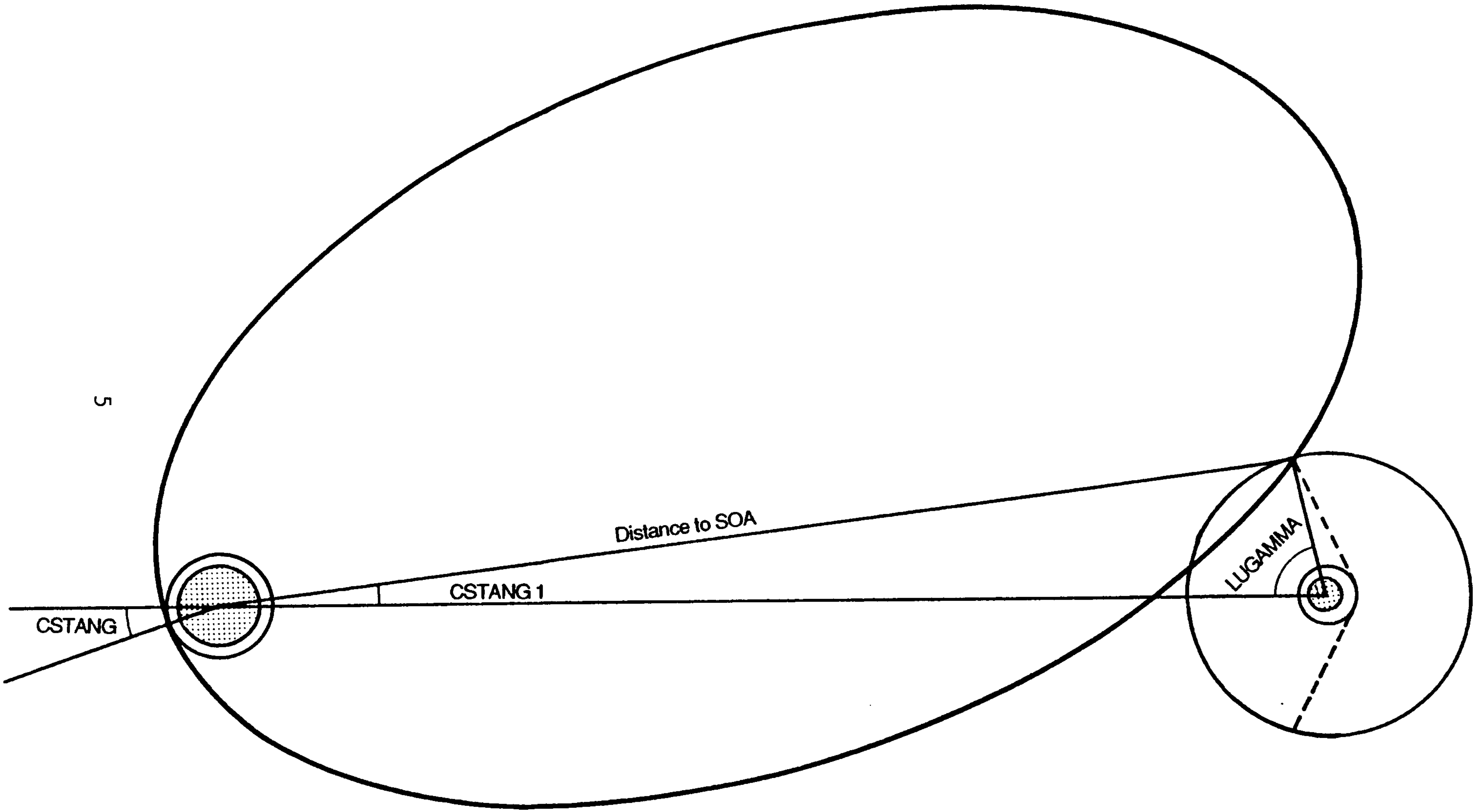


Figure 2

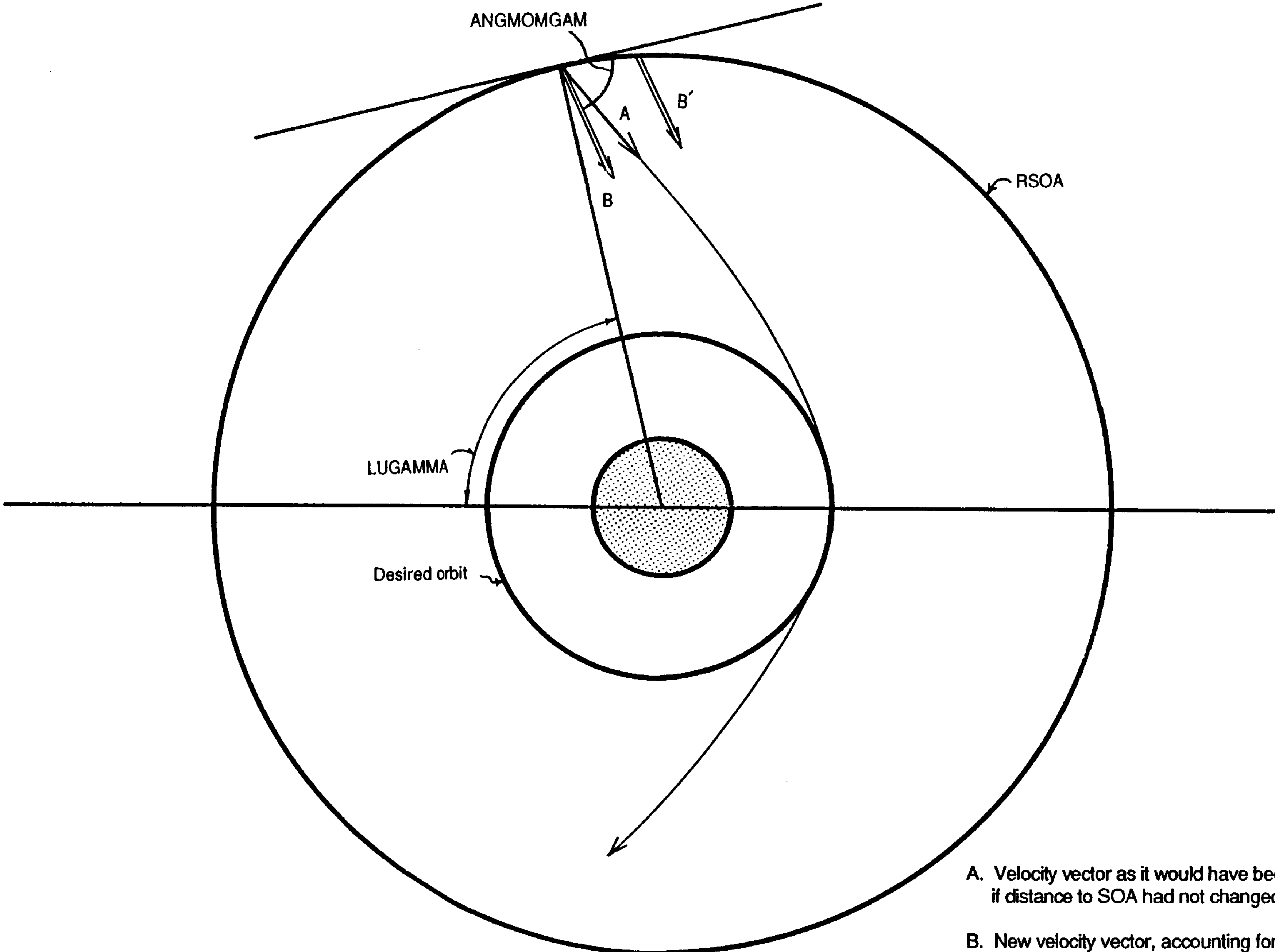


Figure 3

A. Velocity vector as it would have been if distance to SOA had not changed.

B. New velocity vector, accounting for increased distance to SOA.

B′. Desired position of new velocity vector, to intercept orbit perigree.

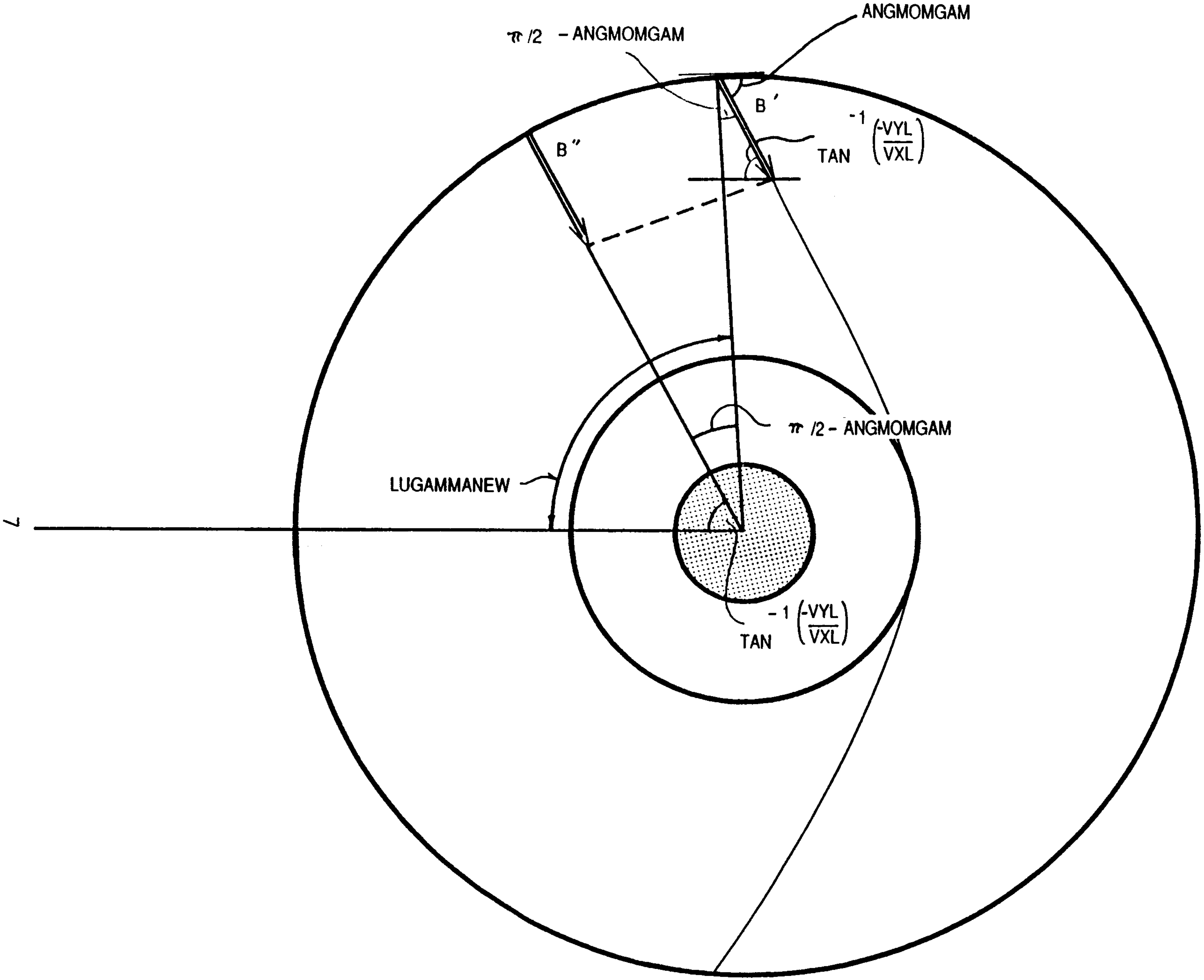


Figure 4

B'=B"

LUGAMMANEW= $\pi/2$ - ANGMOMGAM + $TAN^{-1}\left(\frac{-VYL}{VXL}\right)$

4.0 Program Execution Instructions

The following instructions describe the steps to be taken by the user to execute this program.

A. Obtain access to the DEC VAX minicomputer and sign on with user identification.

B. At the $ prompt, type SETDRV 415 if using a Techtronix 4115 terminal, or SETDRV 407 if using a Techtronix 4111 terminal. If a terminal different from these is being used, consult the system administrator for the correct device driver code.

C. At the next $ prompt, type SETTEK LLOFX if using a Techtronix terminal, or RUN LLOFX if using a different terminal.

D. When prompted by the program, enter the following information:

1. Earth-orbiting space station altitude (kilometers above the Earth's surface).

2. Desired orbital altitude above the Lunar surface (kilometers).

3. Earth-Moon distance at time of Lunar intercept (kilometers). This can range from 359,856 to 405,970 kilometers. The average distance is 384,400 kilometers.

4. Beginning reference point for a screen plot of the transfer orbit (kilometers). The Earth appears at the left side of the screen, and is centered at zero kilometers. The Moon appears at the right side of the screen, and is centered at the range specified in (3) above. The beginning reference point defines the left boundary of the plot. A negative beginning reference point ensures the entire Earth orbit is included in the plot.

5. Ending reference point for the screen plot of the transfer orbit (kilometers). This defines the right boundary of the plot. It must be at least as large as the range specified in (3) above if the Moon is to be included in the plot.

6. Aspect ratio (Y:X ratio) of the screen during the plot of the transfer orbit. This allows adjustment for the fact that a screen pixel is longer than it is wide, and removes the resulting distortion. The valid range for aspect ratio is 0.55 to 0.75.

E. After the aspect ratio has been entered, the program will execute. The following information is displayed upon completion of execution:

1. A message describing whether the transfer orbit is hyperbolic or elliptical.

2. Trans-Lunar injection delta-v.

3. Lunar orbit insertion delta-v.

4. Trans-Earth injection delta-v.

5. Earth orbit insertion delta-v.

6. Total delta-v.

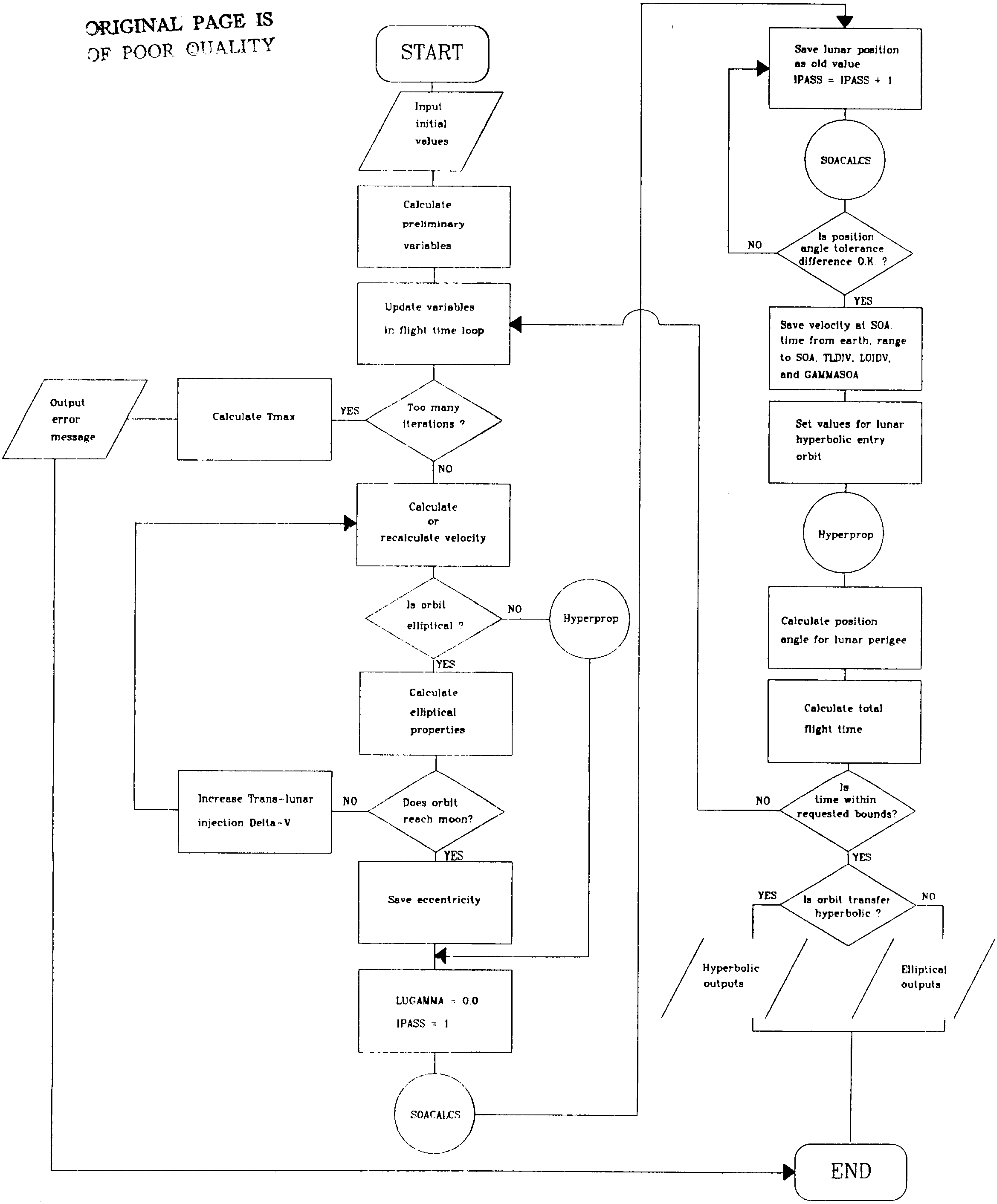
7. Actual time of flight for the outbound leg (Earth to Moon). Return time is the same.

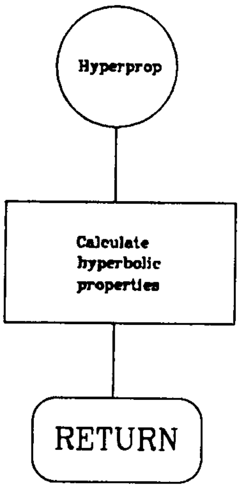
8. A screen plot of the transfer orbit.

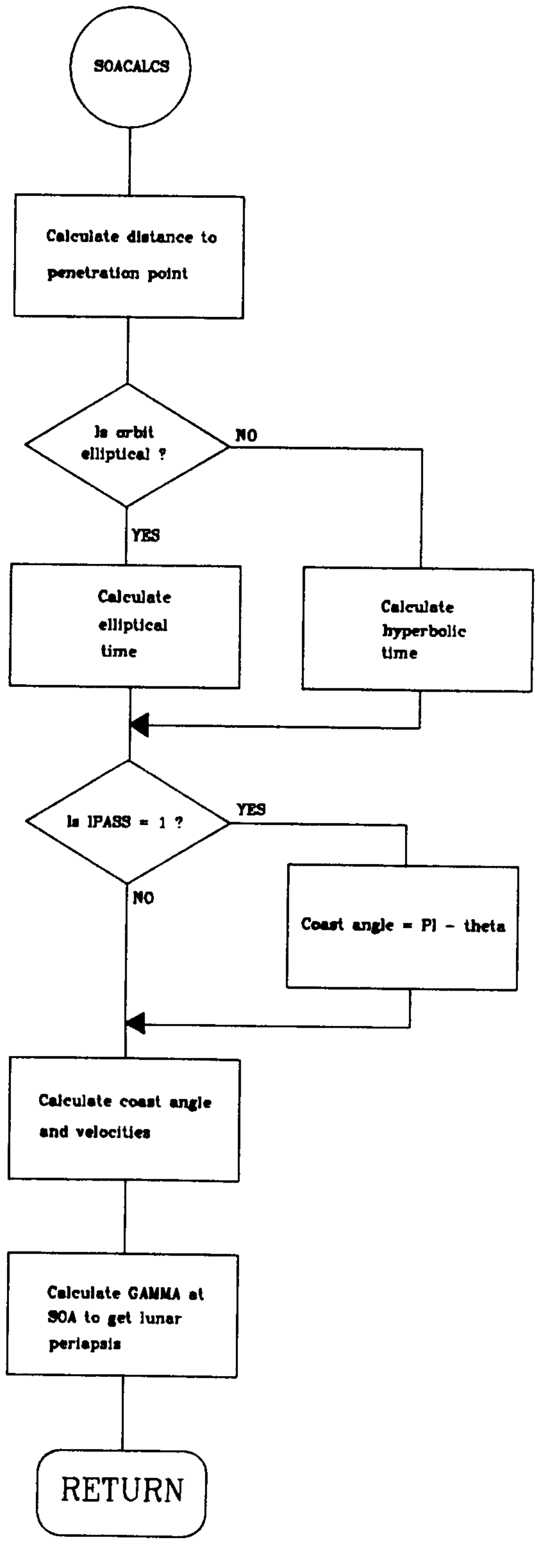
F. To re-execute the program with new parameters, begin again at step (C) above.

Appendix A - Program Flow Chart

ORIGINAL PAGE IS
OF POOR QUALITY




Hyperprop
Calculate hyperbolic properties
RETURN


SOACALCS
Calculate distance to penetration point
Is orbit elliptical ?
NO
YES
Calculate elliptical time
Calculate hyperbolic time
Is IPASS = 1 ?
YES
NO
Coast angle = PI - theta
Calculate coast angle and velocities
Calculate GAMMA at SOA to get lunar periapsis
RETURN

Appendix B - Program Code Listing

```
            PROGRAM
     THIS PROGRAM PROPAGATES ORBIT TO MOON AND BACK
           IN METRIC VECTOR UNITS

     PROGRAM WAS:
              PROPOSED BY GUS BABB
              WRITTEN BY CHRIS VARNER AND MIKE D'ONOFRIO
              DOCUMENTED BY STEVE ERICKSON
              FOR NASA's ADVANCED SPACE TRANSPORTATION SYSTEM
              CONTRACT NO. NAS 9-17878
              EAGLE ENGINEERING INC. 1988

         FIRST DECLARE VARIABLES REAL EXCEPT VARIABLES
               BEGINNING WITH 'I'

     IMPLICIT REAL*4 (A-H,J-Z)
     REAL*4 X(250),Y(250),XE(250),YE(250),XL(250),YL(250)

     OPEN GRAPHICS SUBROUTINES

     CALL JBEGIN
     CALL JDINIT(1)
     CALL JDEVON(1)
     CALL JIENAB(1,4,1)

     INPUTS
     WRITE (5,5)
     FORMAT ('1INPUT ALL NUMBERS AS REAL VALUES')
     WRITE (5,10)
     FORMAT ('0INPUT SPACE STATION ALTITUDE km')
     READ *,H
     WRITE (5,15)
     FORMAT ('0INPUT LUNAR ALTITUDE km')
     READ *,HL
     WRITE (5,17)
     FORMAT ('0INPUT EARTH-MOON DISTANCE avg.=384400')
     READ *,EMRANGE
     WRITE (5,20)
     FORMAT ('0INPUT FLIGHT TIME IN HOURS')
     READ *,OFTHR         WRITE (5,23)
     FORMAT ('0INPUT END X POINT TO VIEW',/,' FROM 0 TO 500000')
     READ *,XMAX
     WRITE (5,24)
     FORMAT ('0INPUT ASPECT RATIO OF SCREEN (Y/X)',/,' FROM
+     .55 TO .75')
     READ *,AR
     XMAX1=.2*(XMAX-XMIN)+XMAX
     XMIN1=XMIN-.2*(ABS(XMAX-XMIN))
     YMIN=(XMIN-XMAX)/2.0
     YMAX=(XMAX-XMIN)/2.0
     YMIN1=(XMIN1-XMAX1)/2.0
     YMAX1=(XMAX1-XMIN1)/2.0
```

```
FUDGE FACTOR COMPENSATING FOR ASPECT RATIO OF SCREEN

YMAX=AR*YMAX
YMAX1=YMAX1*AR
ECCE=0.0
OBTCALC=10000000.0
TIMEERROR=3601.0
TLIDV=3.07

FUDGE FACTOR COMPENSATING FOR ASPECT RATIO OF SCREEN[B

OBFLTTM=OFTHR*3600.
RTNFLTTM=RFTHR*3600.
GAMAIN =0.0
PI=3.1415926535
RO=6378.1
ROL = 1738.0
MUE=398603.0
MUL=4902.97
R=H+RO
VOM=1.02
RPE=R
V=SQRT(MUE/RPE)+TLIDV
ICOUNT=0
RSOA=EMRANGE/10.017
LOOPFLAG$='ON'

START THE FLIGHT TIME LOOP

DO 1400 WHILE(LOOPFLAG$.EQ.'ON')
  GAIN=1./(OBTCALC*10.0)
  TLIDV=TLIDV+TIMEERROR*GAIN
  MU=MUE
  ICOUNT=ICOUNT +1
  R=H+RO
  RPE=R
  RPL=ROL+HL
  GAMMA=PI*GAMAIN/180.

  IS THE ORBIT ELLIPTICAL OR HYPERBOLIC?

  IF(ICOUNT.LT.10000)THEN
    V=SQRT(MUE/RPE)+TLIDV
    IF(V**2.0.LE.(2.*MU/R)) THEN

      ORBIT IS ELLIPTICAL

      Q=R*V**2./MU
      RP=(1.-SQRT(1.-Q*(2.-Q)*(COS(GAMMA))**2))*R/(2.-Q)
      RA=(1.+SQRT(1.-Q*(2.-Q)*(COS(GAMMA))**2))*R/(2.-Q)
      A=(RA+RP)/2.
      ECC=(RA-RP)/(RA+RP)
      P=(1.+ECC)*RP
      NU=SQRT(MU/A**3)
```

```
  COFO=(P/R-1.)/ECC
  FO=ATAN(SQRT(1.0001-COFO**2)/COFO)
  IF (FO.LT.0.) FO=PI+FO
  IF (GAMMA.LT.0.) FO=-FO
  EO=2.*(ATAN(SQRT((1.-ECC)/(1.+ECC))*TAN(FO/2.)))
  THETAO=FO
  TAUO=(EO-ECC*SIN(EO))/NU
  TIMEO=TAUO

  DOES THE ELLIPSE REACH THE MOON?

  IF (RA.LT.(EMRANGE+RSOA)) THEN

    NO

    TLIDV=TLIDV + .01
    GO TO 250
  ENDIF
ELSE

  ORBIT IS HYPERBOLIC

  CALCULATE HYPERBOLIC PROPERTIES

  CALL HYPERPROP(PI,A,B,MU,R,V,GAMMA,P,ECC,RP,
+      VP,COTHETA,THETAO,TIMEO)
ENDIF
LUGAMMA=0.0
IPASS=1
CALL SOACALCS(PI,VXL,IPASS,VYL,VT,GAMMAT,VPL,NU,
+ ANGMOMGAM,RSOA,TFROMPER,RT,MUL,RPL,ECC,P,MU,A,VOM,
+ LUGAMMA,LUGAMMANEW,EMRANGE,TAUO,CSTANG)

ITERATE

  IPASS=IPASS+1

  SAVE THE LUNAR POSITION ANGLE AS AN OLD VALUE

  LUGAMMA=LUGAMMANEW
  CALL SOACALCS(PI,VXL,IPASS,VYL,VT,GAMMAT,VPL,NU,
+ ANGMOMGAM,RSOA,TFROMPER,RT,MUL,RPL,ECC,P,MU,A,VOM,
+ LUGAMMA,LUGAMMANEW,EMRANGE,TAUO,CSTANG)
  IF(ABS(LUGAMMANEW-LUGAMMA).GE..02) GOTO 200

THEN DO ANOTHER ITERATION
SAVE VELOCITY AT SOA, TIME FROM EARTH TO SOA, RANGE TO SOA
TLIDV, AND LOIDV, AND GAMMASOA AS OLD VALUES

TTSOA=TFROMPER
RESOA=RT
TLIDV=V-SQRT(MUE/(H+RO))
LOIDV=VPL-SQRT(MUL/RPL)
AE=A
```

```
      NUE=NU
      PE=P
      ECCE=ECC
      VESOA=VT
      GAMMASOA=GAMMAT

      SET VALUES FOR LUNAR HYPERBOLIC ENTRY ORBIT
      MU=MUL
      V=SQRT(VYL**2+VXL**2)
      GAMMA=ANGMOMGAM
      R=RSOA
      CALL HYPERPROP(PI,A,B,MU,R,V,GAMMA,P,ECC,RP,VP,
 +    COTHETA,THETAO,TIMEO)

      CALCULATE THE POSITION ANGLE FOR LUNAR PERIGEE
      THETALP=LUGAMMA+THETAO
      CALCULATE THE TOTAL FLIGHT TIME AND
      TIME INSIDE SOA
      TIMESOATLOI=TIMEO
      OBTCALC= (ABS(TTSOA)+ABS(TIMESOATLOI))

      CHECK TO SEE IF TIME IS WITHIN REQUESTED BOUNDS
      TIMEERROR=OBTCALC-OBFLTTM
      IF (ABS(TIMEERROR).LE.600.) THEN
        OUTPUTS
        IS IT HYPERBOLIC?
        IF (ECCE.GE.1.0) THEN
          YES
          WRITE(5,50)
0         FORMAT('-__________HYPERBOLIC ORBIT__________')
        ELSE
          WRITE(5,55)
5         FORMAT('-__________ELLIPTICAL ORBIT__________')
        ENDIF
        WRITE(5,60) TLIDV,LOIDV
        FORMAT('0OUTBOUND PROPERTIES',/,' TLI DELTA-V, km/sec.=',
 +      F6.2,/,' LOI DELTA-V, km/sec.=',F6.2)
        WRITE(5,62) LOIDV,TLIDV
        FORMAT(' INBOUND PROPERTIES',/,' TEI DELTA-V= ',F6.2,/,
 +      ' EOI DELTA-V = ',F6.2)
        WRITE(5,65) TLIDV+LOIDV,(ABS(TTSOA)+ ABS(TIMESOATLOI))/3600.
        FORMAT(' TOTAL DELTA-V =',F7.2,/,' FLIGHT TIME = ',F10.3)
        ECCM=ECC

        GRAPHICS

        IT=200
        ECC=ECCE
        A=AE
        P=PE
        MU=MUE
        NU=NUE
        RI=6371.23
        CALL GATTRI (1,0,1.0)
```

```
      CALL GATTRI  (2,0,1.0)
      CALL GATTRI  (3,0,1.0)
      CALL GATTRI  (4,5,1.0)
      CALL GATTRI  (5,5,1.0)
      CALL GATTRI  (6,5,1.0)
      CALL GATTRI  (7,5,1.0)
      CALL GATTRI  (11,5,1.0)
      CALL GATTRI  (12,5,1.0)
      CALL GCHART  (1,5,'EARTH-MOON ORBIT',16)
      CALL GAXIS (1,0,XMIN1,XMAX1,0,
 +    'DISTANCE WITH RESPECT TO EARTH CENTER',
 +    37,YMIN1,YMAX1,0,'KM',2)
      DO 1209 IE=1,3
        ICO=0
        DO 1009 IN=1,IT
          INM1=IN-1
          X(IN)=RI*COS(FLOAT(INM1)*10.*PI/180.)
          Y(IN)=RI*SIN(FLOAT(INM1)*10.*PI/180.)
          IF(IE.NE.1) X(IN)=X(IN)+EMRANGE
          IF(X(IN).GT.XMIN.AND.X(IN).LT.XMAX) THEN
            ICO=ICO+1
            Y(ICO)=Y(IN)
            X(ICO)=X(IN)
          ENDIF
9       CONTINUE

        GRAPH

        CALL JOPEN
        CALL JCOLOR(6)
        IF(IE.EQ.2) CALL JCOLOR(1)
        IF(IE.EQ.3) THEN
          CALL JCOLOR(0)
          CALL GCURVE(X,Y,ICO,-1,0,0)
        ENDIF
        IF(IE.NE.3) CALL GCURVE(X,Y,ICO,0,0,0)
        CALL JCLOSE
        IF(IE.EQ.1) RI=1739.35
        IF(IE.EQ.2) RI=RSOA
9     CONTINUE
      ITIMESE=200
      IXE=0
      DO 1309 I=0,ITIMESE
        IP1=I+1
        RT=(RO+H+1.0)+FLOAT(I)*(RESOA-RO-H)/FLOAT(ITIMESE)
        CALL ORBITPROP(ECC,RT,P,PI,MU,A,THETA,TFROMPER,
 +                     TAUO,GAMMAT,NU,VT)
        XE(IP1)=RT*(-1.0)*COS(CSTANG+THETA)
        YE(IP1)=RT*(-1.0)*SIN(CSTANG+THETA)
        IF  (XE(IP1).GT.XMIN.AND.XE(IP1).LT.XMAX) THEN
          IXE=IXE+1
          XE(IXE)=XE(IP1)
          YE(IXE)=YE(IP1)
        ENDIF
```

```
309        CONTINUE
           CALL JOPEN
           CALL JCOLOR(2)
           CALL GCURVE(XE,YE,IXE,0,0,0)
           CALL JCLOSE
           ECC=ECCM
           MU=MUL
           V=SQRT(VYL**2+VXL**2)
           GAMMA=ANGMOMGAM
           R=RSOA
           CALL HYPERPROP(PI,A,B,MU,R,V,GAMMA,P,ECC,RP,
   +       VP,COTHETA,THETAO,TIMEO)
           THETALP=LUGAMMA+THETAO
           TIMESOATLOI=TIMEO
           ITIMES=31
           ITIMESM1=ITIMES -1
           IXL=0
           DO 1359 I=0,ITIMESM1
             IP1=I+1
             RT=RSOA-FLOAT(I)*(RSOA-RP)/(FLOAT(ITIMESM1))
             CALL ORBITPROP(ECC,RT,P,PI,MU,A,THETA,TFROMPER,
   +         TAUO,GAMMAT,NU,VT)
             XL(IP1)=EMRANGE-RT*COS(THETALP-THETA)
             YL(IP1)=RT*SIN(THETALP-THETA)
             IF(XL(IP1).GT.XMIN.AND.XL(IP1).LT.XMAX) THEN
               IXL=IXL+1
               XL(IXL)=XL(IP1)
               YL(IXL)=YL(IP1)
             ENDIF
359        CONTINUE
           CALL JOPEN
           CALL JCOLOR(4)
           CALL GCURVE(XL,YL,IXL,0,0,0)
           CALL JCLOSE
           INBOUND
           DO 1379 I=1,IXE
             YE(I)=YE(I)*(-1.0)
379        CONTINUE
           DO 1389 I=1,IXL
             YL(I)=YL(I)*(-1.0)
389        CONTINUE
           CALL JOPEN
           CALL JCOLOR(2)
           CALL GCURVE(XE,YE,IXE,0,0,0)
           CALL JCOLOR(4)
           CALL GCURVE(XL,YL,IXL,0,0,0)
           CALL JCLOSE
           LOOPFLAG$='OFF'
         ENDIF
       ELSE
         PRINT *,'***************LOOPING ERROR OCCURED********'
         PRINT *,'PROBABLY CAUSED BY EXCESSIVE FLIGHT TIME'
         PRINT *,'MAXIMUM FLIGHT TIME IS APPROX 75 HRS AT LUNAR'
         PRINT *,'PERIGEE AND 120 HRS AT LUNAR APOGEE'
```

```
          LOOPFLAG$='OFF'
        ENDIF
)0    CONTINUE
      CALL JPAUSE(1)
      CALL JDEVOF(1)
      CALL JDEND(1)
      CALL JEND
      STOP
      END


        SUBROUTINE HYPERPROP(PI,A,B,MU,R,V,GAMMA,P,ECC,RP,
     +                       VP,COTHETA,THETAO,TIMEO)


          IMPLICIT REAL*4(A-Z)
          A=MU*R/(R*V**2-2.0*MU)
          B=SQRT(R**3*V**2*(COS(GAMMA))**2/(R*V**2-2.0*MU))
          P=R**2*V**2*COS(GAMMA)**2/MU
          ECC=SQRT(P/A+1.0)
          RP=A*(ECC-1.0)
          VP=SQRT(MU/P)*(1.0+ECC)
          COTHETA=(P/R-1.0)/ECC
          THETAO=ATAN(SQRT(1.0001-COTHETA**2)/COTHETA)
          IF(THETAO.LT.0.0) THETAO=PI+THETAO
          IF(GAMMA.LT.0.0) THETAO=-THETAO
          TIMEO1=SQRT(P**3/MU)/(ECC**2-1.0)
          TIMEO2=ECC*SIN(THETAO)/(1.0+ECC*COTHETA)
          TIMEO3=(1.0/SQRT(ECC**2-1.0))*LOG((ECC+COTHETA+SQRT(
     +           ECC**2-1.0)*SIN(THETAO))/(1.0+ECC*COTHETA))
          TIMEO=TIMEO1*(TIMEO2-TIMEO3)
          RETURN
          END


         SUBROUTINE SOACALCS(PI,VXL,IPASS,VYL,VT,GAMMAT,VPL,NU,
     +                       ANGMOMGAM,RSOA,TFROMPER,RT,MUL,RPL,ECC,
     +                       P,MU,A,VOM,LUGAMMA,LUGAMMANEW,EMRANGE,
     +                       TAUO,CSTANG)


            IMPLICIT REAL*4(A-H,J-Z)
            DL=RSOA*COS(LUGAMMA)
            DE=EMRANGE-DL
            RT=SQRT(DE**2+RSOA**2-DL**2)
            CALL ORBITPROP(ECC,RT,P,PI,MU,A,THETA,TFROMPER,
     +                     TAUO,GAMMAT,NU,VT)
            RY=RSOA*SIN(LUGAMMA)
            CSTANG1=ATAN(RY/DE)
            CSTANG=PI+CSTANG1-THETA
            VXE=VT*SIN(GAMMAT+PI-CSTANG-THETA)
            VYE=VT*COS(GAMMAT+PI-CSTANG-THETA)
```

```
      VXL=VXE
      VYL=VYE-VOM

      CALCULATE GAMMA AT SOA TO GET LUNAR PERIAPSIS

      VSOAL2=VXL**2 + VYL**2
      VPL2= VSOAL2-(2.0*MUL/RSOA)+(2.0*MUL/RPL)
      VSOAL=SQRT(VSOAL2)
      VPL=SQRT(VPL2)
      ANGMOMGAM=ATAN(SQRT(VSOAL2*RSOA**2-VPL2*RPL**2)/
+                (VPL*RPL))
      LUGAMMANEW=ATAN(-VYL/VXL)-ANGMOMGAM +PI/2.
      RETURN
      END



   SUBROUTINE ORBITPROP(ECC,RT,P,PI,MU,A,THETA,TFROMPER,
+                        TAUO,GAMMAT,NU,VT)


      IMPLICIT REAL*4(A-H,J-Z)
      IF(ECC.LT.1.0) THEN

        ELLIPTICAL ORBIT
        PROPAGATE ELLIPTICAL ORBIT FORWARD TO RT OUTBOUND

        F=ATAN((SQRT(ECC**2*RT**2-(P-RT)**2))/(P-RT))
        IF(F.LT.0.0) F=F+PI
        VT=SQRT(MU*(2.0/RT-1.0/A))
        GAMMAT=ATAN(ECC*SIN(F)/(1.0+ECC*COS(F)))
        ETH=2.0*(ATAN(SQRT((1.0-ECC)/(1.0+ECC))*TAN(F/2.0)))
        TAUTH=(ETH-ECC*SIN(ETH))/NU
        TFROMPER=TAUTH-TAUO
        THETA=F
      ELSE
        HYPERBOLIC ORBIT
        ORBIT PROPAGATION FOR HYPERBOLIC ORBIT
        COTHETA=(P/RT-1.0)/ECC
        THETA=ATAN(SQRT(1.0001-COTHETA**2)/COTHETA)
        IF(THETA.LT.0.0) THETA=PI+THETA
        VT=SQRT(MU*(2.0/RT+1.0/A))
        COGAMAT=SQRT(P*MU)/(RT*VT)
        GAMMAT=ATAN(SQRT(1.0001-COGAMAT**2)/COGAMAT)
        IF(THETA.LT.0.0) GAMMAT=-GAMMAT
        TFROMPER1=SQRT(P**3/MU)/(ECC**2-1.0)
        TFROMPER2=ECC*SIN(THETA)/(1.0+ECC*COTHETA)-(1.0/SQRT(
+                 ECC**2-1.0))*LOG((ECC+COTHETA+SQRT(ECC**2-1.0)
+                 *SIN(THETA))/(1.0+ECC*COTHETA))
        TFROMPER=TFROMPER1*TFROMPER2
      ENDIF
      RETURN
      END
```

Appendix C - Program Variables

VARIABLE	DESCRIPTION
A	Semimajor Axis of the Transfer Orbit (km)
AE	Semimajor Axis of the Transfer Orbit (km), Earth coordinates. Stored Value of "A" Before Switching to Lunar Coordinates.
AR	Aspect Ratio of Screen Plot
ANGMOMGAM	Flight Path Angle at Sphere of Action Penetration Point for the Lunar Hyperbolic Orbit (rad)
B	Hyperbolic Asymptotes Parameter
COFO	Cosine of the True Anomoly (dimensionless)
COGAMAT	Cosine of the Transfer Orbit Flight Path Angle at the SOA (non-dimensional)
COTHETA	Cosine of the True Anomoly for Hyperbolic Orbits (non-dimensional)
CSTANG	Actual Coast Angle (rad)
CSTANG1	Angle from the Earth-Moon Line to the Sphere of Action Penetration Point (rad)
CSTANGO	Coast Angle Initial Guess (rad)
DE	X Component of Distance from the Earth to Sphere of Action Penetration Point (km)
DL	X Component of Distance from the Moon to Sphere of Action Penetration Point, Measured Positive in the Negative X Direction (km)
ECC	"New" Eccentricity of the Transfer Orbit (dimensionless)
ECCE	"Old" Eccentricity of the Transfer Orbit (dimensionless)
ECCM	Eccentricity of the Hyperbolic Lunar Fly-By Orbit
EMRANGE	Earth-Moon Range (km)
EO	Initial Eccentric Anomoly Along the Transfer Orbit (rad)
ETH	Eccentric Anomoly for the Point of Sphere of Action Penetration Along the Transfer Orbit (rad)

F	True Anomoly for the Intersection Point of the Transfer Orbit and the Lunar Sphere of Action (rad)
FO	Initial True Anomoly Along the Transfer Orbit (rad)
GAIN	Error Multiplier for Adjusting Initial Velocity Change for a New Flight Time (delta-v/s)
GAMAIN	Inertial Flight Path Angle at Transfer Orbit Perigee (deg)
GAMMA	Inertial Flight Path Angle at Transfer Orbit Perigee (rad)
GAMMASOA	Final Inertial Flight Path Angle of the Transfer Orbit at the Sphere of Action Penetration Point (rad)
GAMMAT	Inertial Flight Path Angle of the Transfer Orbit at Sphere of Action Penetration (rad)
H	Space Station Orbital Altitude (km)
HL	Lunar Orbit Altitude (km)
I	Iteration Counter Used During Transfer and Lunar Fly-By Orbit Plots
ICO	Plot Matrix Subscript Used to Store Circle Plot Position Values Within the User-Specified Range
ICOUNT	Iteration Counter
IE	Identifier Used During Circle Plotting to Determine If Circle is the Earth, the Moon, or the SOA
IN	Iteration Counter Used During Circle Plotting Routines
INM1	IN Minus 1
IP1	I Plus 1
IPASS	Iteration Counter
IT	Constant. Maximum Value of IN
ITIMES	Number of Plot Positions in Lunar Fly-By Orbit
ITIMESE	Number of Plot Positions in Transfer Orbit
ITIMESM1	ITIMES Minus 1
IXE	Plot Matrix Subscript Used to Store Transfer Orbit Plot Position Values Within the User-Specified Range
IXL	Plot matrix Subscript Used to Store Hyperbolic Lunar Fly-By Orbit Plot Position Values Within the User-Specified Range

LOIDV	Lunar Orbit Injection Velocity Change (km/s)
LOOPFLAG	Do-While Flag That Limits the Number of Iterations Seeking Correct Flight Time
LUGAMMA	Angle Between the Earth-Moon Line and the "Old" Sphere of Action Penetration Point (rad)
LUGAMMANEW	Angle Between the Earth-Moon Line and the "New" Sphere of Action Penetration Point (rad)
MU	Local Gravity Constant (km^3/s^2)
MUE	Gravitational Constant for the Earth (km^3/s^2)
MUL	Gravitational Constant for the Moon (km^3/s^2)
NU	Mean Motion of the Transfer Orbit (rad/s)
NUE	Stored Value of NU. Recalled During Plotting Routines
OBFLTTM	Outbound Flight Time (secs)
OBTCALC	Total Time of Flight from Earth Orbit to Lunar Orbit (sec)
OFTHR	Outbound Flight Time (hrs)
P	Semi-Latus Rectum of the Transfer Orbit (km)
PE	Stored Value of P. Recalled During Plotting Routines
Q	Vis-Viva Parameter (dimensionless)
R	Orbital Radius of the Space Station (km)
RA	Radius to the Apogee of the Transfer Orbit (km)
RESOA	Final Distance to the Sphere of Action (km)
RFTHR	Return Flight Time (hrs)
RI	Various Circle Radii, Used During Plotting Routines
RO	Radius of the Earth (km)
ROL	Radius of the Moon (km)
RP	Radius to the Perigee of the Transfer Orbit (km)
RPE	Perigee Radius at the Earth (km)
RPL	Perigee Radius at the Moon (km)

RSOA	Radius of the Lunar Sphere of Action (km)
RT	Range to Lunar Sphere of Action (km)
RTNFLTTM	Return Flight Time (secs)
RY	Y Component of the Distance from the Moon to the Sphere of Action Penetration Point (km)
TAUO	Initial Time from Perigee (sec)
TAUTH	Time of Sphere of Action Penetration from Transfer Orbit Perigee (sec)
TFROMPER	Time of Sphere of Action Penetration from TLI (sec)
TFROMPER1	Intermediate Calculation for TFROMPER
TFROMPER2	Intermediate Calculation for TFROMPER
THETA	Position Angle for the Sphere of Action Penetration Point, Earth Reference (rad)
THETALP	Lunar Perigee Position Angle (rad)
THETAO	Initial Position Angle, and True Anomoly for Hyperbolic Orbits (rad)
TIMEERROR	Difference Between Total Time of Flight and Requested Time of Flight (sec)
TIMEO	Initial Time From Perigee (sec)
TIMEO1	Intermediate Calculation for TIMEO
TIMEO2	Intermediate Calculation for TIMEO
TIMEO3	Intermediate Calculation for TIMEO
TIMESOATLOI	Time from Sphere of Action Penetration to Lunar Orbit Injection (sec)
TLIDV	Trans Lunar Injection Velocity Change (km/s)
TMAX	Approximate Maximum Flight Time Allowed (hrs)
TTSOA	Final Time to the Sphere of Action (sec)
V	Velocity (km/s)
VESOA	Inertial Velocity at the Sphere of Action Penetration Point (km/s)
VOM	Velocity of the Moon (km/s)

VP	Hyperbolic Velocity at Perigee (km/s)
VPL	Lunar Relative Velocity at the Lunar Orbit Radius (km/s)
VPL2	Square of the Lunar Relative Velocity at the Lunar Orbit Radius (km^2/s^2)
VSOAL	Lunar Relative Velocity at the Sphere of Action (km/s)
VSOAL2	Square of the Lunar Relative Velocity at the Sphere of Action (km^2/s^2)
VT	Velocity at Sphere of Action Penetration (km/s)
VXE	X Velocity Component (Earth Coordinates: Inertial)
VXL	X Velocity Component (Lunar Coordinates)
VYE	Y Velocity Component (Earth Coordinates: Inertial)
VYL	X Velocity Component (Lunar Coordinates)
X	X-Component of Plot Position For Circles
XE	X-Component of Plot Position For Transfer Orbit
XL	X-Component of Plot Position For Hyperbolic Lunar Fly-By Orbit
XMAX	High X-Boundary Value Supplied By User For Screen Plot
XMAX1	Additional 20% Added to XMAX For Plot Border
XMIN	Low X-Boundary Value Supplied By User For Screen Plot
XMIN1	Additional 20% Subtracted From XMIN For Plot Border
Y	Y-Component of Plot Position For Circles
YE	Y-Component of Plot Position For Transfer Orbit
YL	Y-Component of Plot Position For Hyperbolic Lunar Fly-By Orbit
YMAX	High Y-Boundary Value For Screen Plot, Adjusted For Aspect Ratio
YMAX1	Preliminary High Y-Boundary Value For Screen Plot
YMIN	Low Y-Boundary Value For Screen Plot

Appendix D - Calculation of CSTANG

The coast angle (CSTANG) is the angle past the Earth-Moon line at which a vehicle performs a TLI burn, allowing it to intercept the Moon's SOA at the proper point (see Figure D-1).

The radial vector $\vec{r}_A$ locates the point on the transfer ellipse such that this vector's magnitude equals the distance from the Earth to the SOA at the SOA's closest point. F0 is defined to be the true anomoly of $\vec{r}_A$ (see Figure D-2).

The radial vector $\vec{r}_B$ locates the SOA impact point. F1 is defined to be the true anomoly of $\vec{r}_B$. ΔF is defined to be the difference between the anomolies.

$$\Delta F = F1 - F0.$$

The initial coast angle (CSTANG0) is the angle betwween the major axis of the transfer ellipse and $\vec{r}_A$.

$$CSTANG0 = \pi - F0.$$

The angle of SOA impact (CSTANG1) is the angle between the Earth-Moon line and $\vec{r}_B$.

$$CSTANG1 = TAN^{-1}\left[\frac{RSOA * SIN(LUGAMMA)}{EMRANGE - RSOA*COS(LUGAMMA)}\right]$$

From Figure D-1 it can be seen that

$$\begin{aligned} CSTANG &= CSTANG0 + CSTANG1 - \Delta F \\ &= (\pi - F0) + CSTANG1 - (F1 - F0) \\ &= \pi + CSTANG1 - F1. \end{aligned}$$

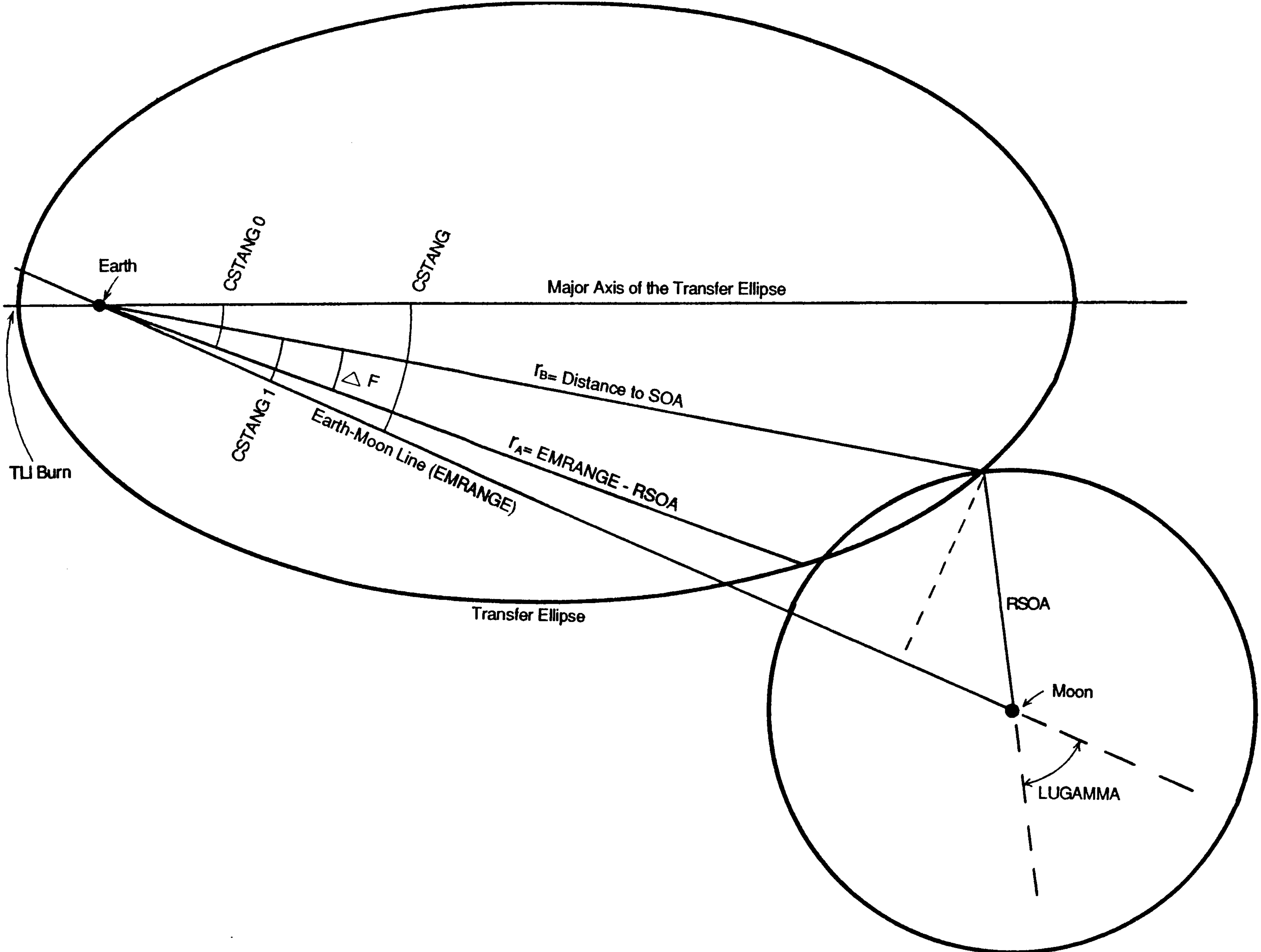

Earth
CSTANG 0
CSTANG
Major Axis of the Transfer Ellipse
△ F
r_B= Distance to SOA
CSTANG 1
Earth-Moon Line (EMRANGE)
r_A= EMRANGE - RSOA
TLI Burn
Transfer Ellipse
RSOA
Moon
LUGAMMA

Figure D-1

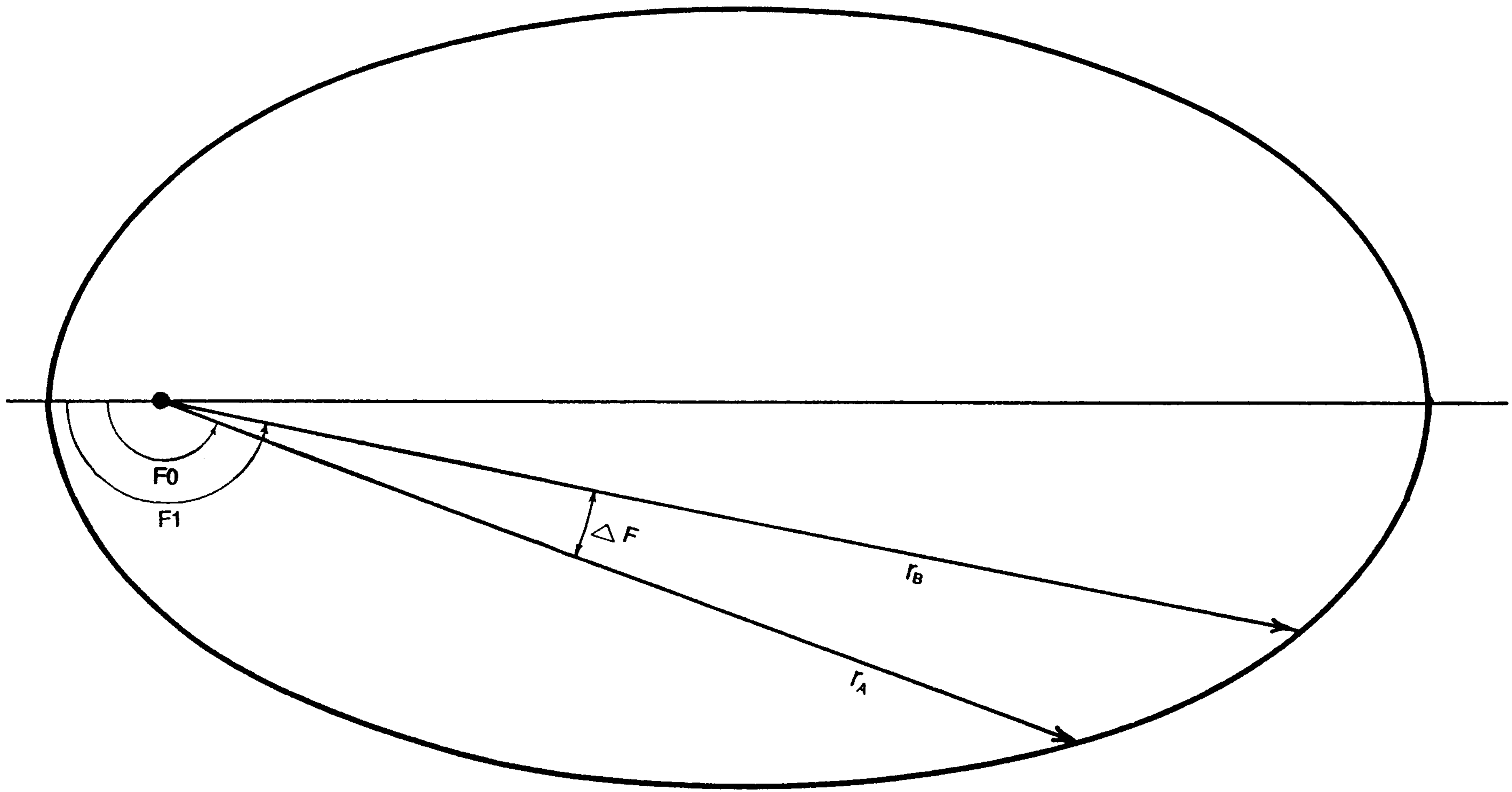


Figure D-2

Appendix E - Calculation of ANGMOMGAM

The flight path angle (γ) of the vehicle at Lunar SOA is calculated in terms of the vehicle's angular momentum ($\vec{h}$). The angular momentum is defined to be the cross-product of the position vector ($\vec{r}$) and the linear momentum vector ($m\vec{v}$), where γ is in the $\vec{r}$, $\vec{v}$ plane and $\vec{h}$ is perpendicular to this plane (see Figures E-1 and E-2).

$$\vec{h} = \vec{r} \text{ X } m\vec{v} = h \cdot \vec{u} = (m \cdot r \cdot v \cdot \cos\gamma) \cdot \vec{u}$$

$$h = m \cdot r \cdot v \cdot \cos\gamma$$

Angular momentum is constant along a given orbit. Therefore, the angular momentum at perigee is the same as the angular momentum at the sphere of action. If the magnitude of the velocity vector is known at the sphere of action, then the velocity at a specified perigee (v_p) can be calculated.

$$v_p = \sqrt{v^2 - v^2_{esc(SOA)} + v^2_{esc(perigee)}}$$

$$= \sqrt{v^2 - 2 \cdot (\mu_M / r_{SOA}) + 2 \cdot (\mu_M / r_p)}$$

where v = magnitude of the velocity vector at SOA
μ_M = moon's gravitational constant
r_{SOA} = radial distance to the SOA
r_p = radial distance to perigee.

At perigee, the flight path angle is zero by definition, and the magnitude of the orbit's angular momentum is the simple product of the distance and velocity.

$$h = m \cdot r_p \cdot v_p$$

Since the angular momentum is the same at the sphere of action, the flight path angle can be calculated.

$$h = m \cdot r_p \cdot v_p = m \cdot r_{SOA} \cdot v \cdot \cos \gamma$$

$$\gamma_{SOA} = \cos^{-1}\left[\frac{r_p \cdot v_p}{r_{SOA} \cdot v_{SOA}}\right] = \text{ANGMOMGAM.}$$

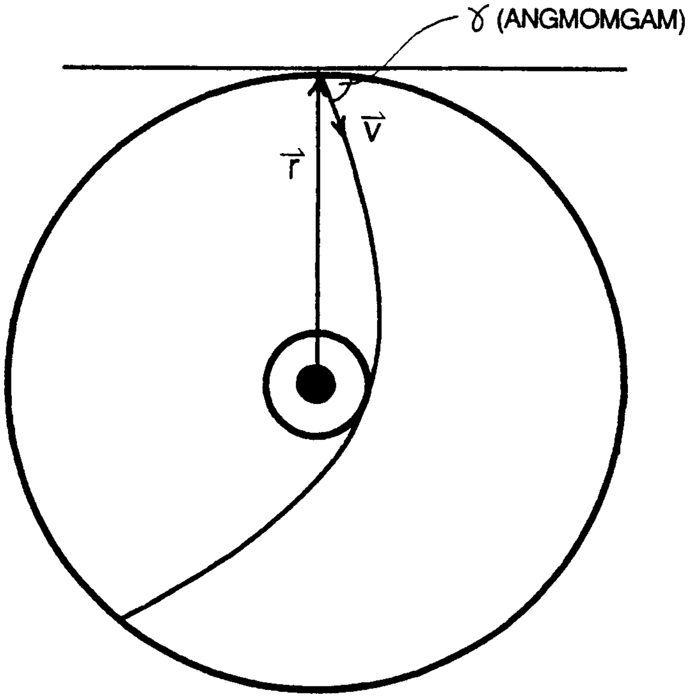


Figure E-1

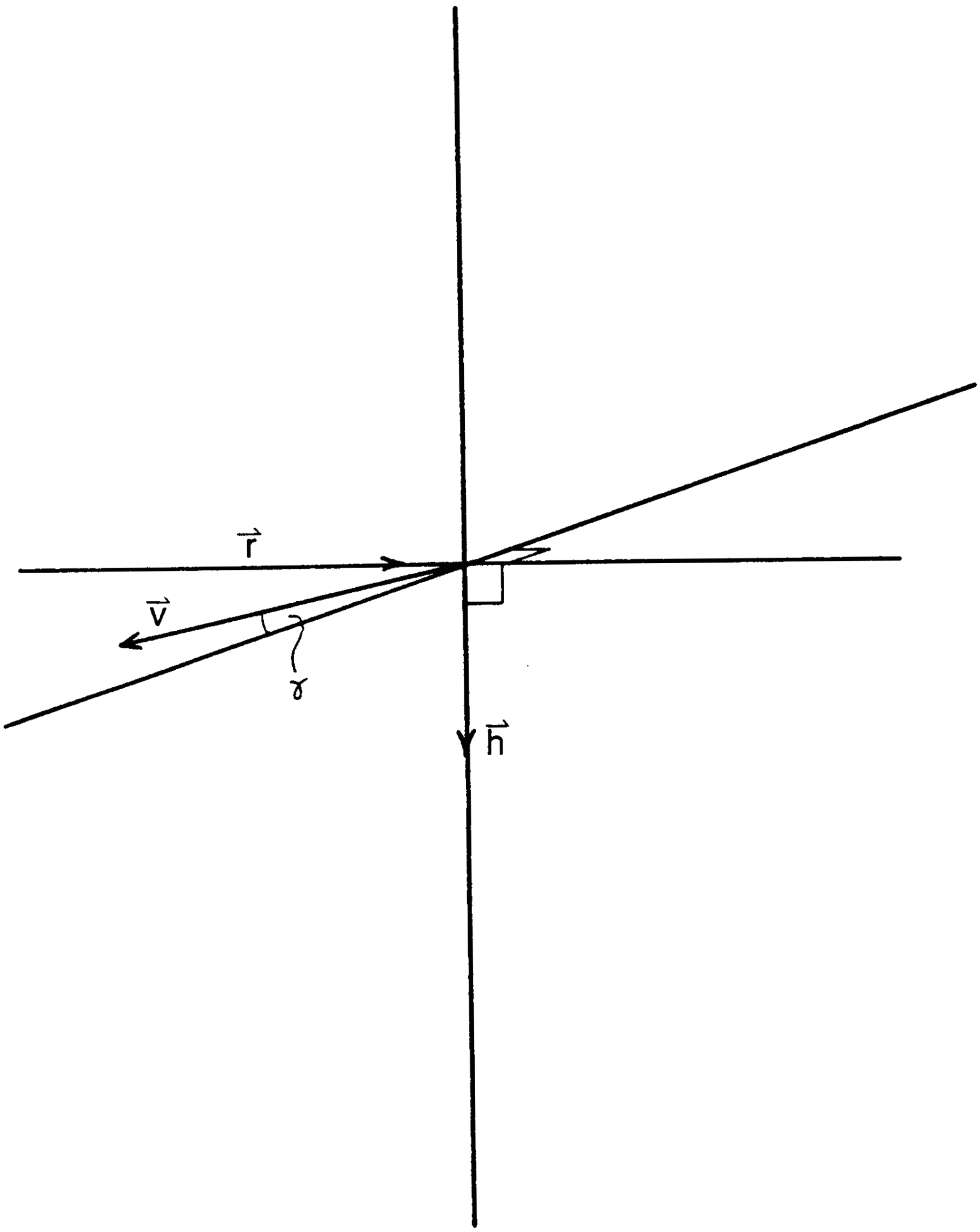


Figure E-2

Appendix F - Test Cases

The attached tables show actual Apollo transfer orbit data compared with the outputs of the program LLOFX. The burns shown in the chart include the Trans-Lunar Injection (TLI), Lunar Orbit Insertion (LOI), and Trans-Earth Injection (TEI). The TLI burn puts the vehicle in a hyperbolic or elliptical orbit to get in the sphere of influence about the moon. Next, the LOI burn puts the vehicle in a circular orbit of 60 nautical miles around the moon. When returning to the Earth a TEI burn is made in lunar holding orbit. The Apollo flights directly entered the Earth's atmosphere to perform a splash down. The program assumes that vehicles will come back to a transportation node in low Earth orbit (i.e. space station). To simulate an Apollo entry, the program LLOFX was run with an Earth holding orbit of 10.16 km. Listed are differences that may cause discrepencies: Apollo used free return trajectories that are not used in LLOFX. Apollo vehicles made plane changes going into transfer orbits and during Lunar Orbit Insertion, while LLOFX does not. Apollo confronted out-of-plane Earth-moon trajectories while LLOFX assumes in-plane. Apollo spacecraft had gravity losses due to long burns versus LLOFX's assumption of instantaneous burns. Also, Earth-moon perturbations might also affect output.

Although many assumptions are made, the numbers are within requested bounds. The TLI burns are less than 5% off the Apollo data and LOI numbers are within approximately 10% of the Apollo data. Therefore, LLOFX is useful for rough performance estimates for lunar vehicles.

OUTBOUND PROPERTIES
Check Cases

Flight #	Earth Moon Distance (km)	Flight Time (hrs)	Earth Orbit Altitude (km)	Lunar Orbit Altitude (km)	TLI Delta-V (km/sec)	LOI Delta-V (km/sec)
Apollo 11	351600	73.09	334.7	111.2	3.18	.89
LLOFX	351600	73.02	334.7	111.2	3.10	.81
Apollo 12	377900	80.63	356.0	111.2	3.20	.88
LLOFX	377900	80.78	356.0	111.2	3.10	.81
Apollo 14	372500	79.47	331.9	107.7	3.16	.92
LLOFX	372500	79.21	331.9	107.7	3.11	.81
Apollo 15	350900	75.68	257.6	106.9	3.17	.91
LLOFX	350900	74.75	257.6	106.9	3.12	.80
Apollo 16	371800	76.00	316.9	108.4	3.17	.85
LLOFX	371800	76.09	316.9	108.4	3.11	.82
Apollo 17	398900	83.00	300.9	108.8	3.12	.79
LLOFX	398900	82.89	300.9	108.8	3.12	.83

INBOUND PROPERTIES
Check Cases

Flight #	Flight Time (hrs)	Earth Altitude (km)	TEI Delta-V (km/sec)
Apollo 11	59.66	10.2	1.00
LLOFX	59.72	10.2	.90
Apollo 12	71.88	10.2	.93
LLOFX	71.88	10.2	.86
Apollo 14	63.22	10.2	1.06
LLOFX	63.30	10.2	.88
Ref. 1			.93
Apollo 15	71.17	10.2	.93
LLOFX	71.25	10.2	.81
Apollo 16	65.27	10.2	1.03
LLOFX	65.31	10.2	.89
Ref. 1			.93
Apollo 17	67.61	10.2	.93
LLOFX	67.68	10.2	.93

REFERENCES

1. Lunar Missions and Exploration, C.T. Leondes ed., Wiley and Sons: New York, New York, 1964.

2. Apollo Mission Reports.